Document made available under the Patent Cooperation Treaty (PCT)

International application number: PCT/SG05/000037

International filing date: 14 February 2005 (14.02.2005)

Document type: Certified copy of priority document

Document details: Country/Office: IN
Number: 337/CHE/2004
Filing date: 15 April 2004 (15.04.2004)

Date of receipt at the International Bureau: 13 May 2005 (13.05.2005)

Remark: Priority document submitted or transmitted to the International Bureau in compliance with Rule 17.1(a) or (b)

World Intellectual Property Organization (WIPO) - Geneva, Switzerland
Organisation Mondiale de la Propriété Intellectuelle (OMPI) - Genève, Suisse

PCT/SG 2005/000037

INTELLECTUAL PROPERTY
INDIA

सत्यमेव जयते

GOVERNMENT OF INDIA
PATENT OFFICE
Ministry of Commerce and Industry
Department of Industrial Policy and Promotion

It is hereby certified that annexed here to is a true copy of **Application, Provisional Specification & Drawings** of the patent application as filed and detailed below:-

Date of application	:	**15-04-2004**
Application No	:	**337/CHE/2004**
Applicants	:	**M/s. Matrixview Technologies (India) Private Limited, No.69, Mahalakshmi Koil Street, Kalakshetra Colony, Besant Nagar, Chennai – 600 090, India an Indian Company**

In witness there of
I have here unto set my hand

Dated this the 12th day of April 2005
22th day of Chaitra, 1926(Saka)

By Authority of
THE CONTROLLER GENERAL OF PATENTS,
DESIGNS AND TRADE MARKS.

M. S. V

(M.S.VENKATARAMAN)
ASSISTANT CONTROLLER OF PATENTS & DESIGNS

PATENT OFFICE BRANCH
Guna Complex, 6th Floor, Annex.II
No.443, Anna Salai, Teynampet,
Chennai – 600 018. India.

Received Rs 3000/- in Cash Cheque / M.O. / P.O. / D.D. / on 15.04.04 Vide C.B.R. No. 7889.

1514

ORIGINAL

337/CHE/2004

15 APR 2004

FORM 1

THE PATENTS ACT,1970
(39 OF 1970)
APPLICATION FOR GRANT OF A PATENT
(SEE SECTIONS 5(2),7,54 AND 135 AND RULE 39)

1. WE, MATRIXVIEW TECHNOLOGIES (INDIA) PRIVATE LIMITED, of NO. 69, MAHALAKSHMI KOIL STREET, KALAKSHETRA COLONY, BESANT NAGAR, CHENNAI -600090, INDIA AN INDIAN COMPANY

2. hereby declare -

(a) that we are in possession of an invention titled "A METHOD AND SYSTEM FOR COMPRESSING IMAGE DATA"

(b) that the Provisional Specification relating to this invention is filed with this application.

(c) that there is no lawful ground of objection to the grant of a Patent to us.

3. We further declare that the inventors for the said invention is/are :

NAME (a)	ADDRESS (b)	NATIONALITY (c)
THIAGARAJAN ARVIND	H24/6, VAIGAI STREET, BESANT NAGAR, 600090 CHENNAI, TAMIL NADU, INDIA	INDIAN

4. That we are assignees of the inventor.

5. That our address for service in India is as follows:- D. P. AHUJA & CO., 53 Syed Amir Ali Avenue, Calcutta 700 019, West Bengal, India. TEL: (033)22819195, FAX: (033)24757524.

6. That to the best of our knowledge, information and belief the fact and matters stated herein are correct and that there is no lawful ground of objection to the grant of patent to us on this application.

7. Following are the attachments with the application:

(a) Provisional Specification (2 copies)
(b) Statement and Undertaking on Form 3 in duplicate
(c) Formal drawings (12 sheets) (Provisional) in duplicate
(d) Rs 3,000/- by cheque bearing No.906542 dated 13.04.2004 on ICICI BANK.

Contd...2

We request that a patent may be granted to us for the said invention.

Dated this 13th day of April, 2004.

(S.D. AHUJA)
OF D. P. AHUJA & CO
APPLICANTS' AGENT

To
The Controller of Patents,
The Patent Office,
Chennai

FORM 2

THE PATENTS ACT, 1970
(39 of 1970)

PROVISIONAL SPECIFICATION
(See Section 10)

TITLE

A METHOD AND SYSTEM FOR COMPRESSING IMAGE DATA

APPLICANT

MATRIXVIEW TECHNOLOGIES (INDIA) PRIVATE LIMITED,
of NO. 69, MAHALAKSHMI KOIL STREET, KALAKSHETRA COLONY,
BESANT NAGAR, CHENNAI -600090,
INDIA
AN INDIAN COMPANY

The following specification particularly describes the nature of the invention

Field of the Invention

The present invention relates to a method and system for compressing image data and other highly correlated data streams.

Background of Invention

Image and data compression is of vital importance and has great significance in many practical applications. To choose between lossy compression and lossless compression depends primarily on the application.

Some applications require a perfectly lossless compression scheme so as to achieve zero errors in the automated analysis. This is particularly relevant when where an automatic analysis is performed on the image or data. Generally, Huffman coding and other source coding techniques are used to achieve lossless compression of image data.

In certain other applications, the human eye visually analyzes images. Since the human eye is insensitive to certain patterns in the images, such patterns are discarded from the original images so as to yield good compression of data. These schemes are termed as "visually lossless" compression schemes. This is not a perfectly reversible process as the de-compressed image data is different from the original image data. The degree of difference depends on the quality of compression, and the compression ratio. Compression schemes based on discrete cosine transforms and wavelet transforms followed by lossy quantization of data are typical examples of visually lossless scheme.

As a general rule, it is desirable to achieve the maximum compression ratio with zero, or minimal, possible loss in the quality of the image. At the same time, the complexity involved in the system and the power consumed by the image compression system are important parameters when it comes to a hardware-based implementation.

Usually, image compression is carried out in two steps. The first step is to use a pre-coding technique, which is normally based on signal transformations. The second step would be to further compress the data values by standard source coding techniques such as, for example, Huffman and Lempel-Ziv schemes.

The initial pre-coding step is the most critical and important operation in image compression. The complexity involved with DCT and Wavelet based transformations is quite high because of the large number of multiplications involved. This is illustrated in the following DCT equation:

$$DCT(i,j)=\frac{1}{\sqrt{2N}}C(i)C(j)\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}f(x,y)\cos\left[\frac{(2x+1)i\pi}{2N}\right]\cos\left[\frac{(2y+1)j\pi}{2N}\right]$$

where $C(x)=\frac{1}{\sqrt{2}}$ if $x=0$, else 1 if $x>0$.

In addition to the large number of multiplications involved in carrying out the above DCT equation, there is also a zigzag rearrangement of the image data, which involves additional complexity. These conventional schemes for image compression are not very well suited for hardware-based implementation.

DCT transformation uses a mathematical algorithm to generate frequency representations of a block of video pixels. DCT is an invertible, discrete orthogonal transformation between time and frequency domain.

Transformation aids in increasing the efficiency of a second step, the entropy coder. At this stage, if the entropy coder produces good compression ratios, then the pre-coding should transform the data into a form suitable for the entropy coder. If the transformation is not efficient, then the entropy coder is becomes redundant. Thus, pre-coding is the most important stage of any image compression algorithm.

Another important property of any transformation is that it is reversible, to allow the reverse process to be applied at the decompression stage to obtain the original image. This transformation is extensively used in JPEG algorithms and their variants.

However, DCT suffers from several problems. Firstly, the complexity of the equation in terms of the number of multiplications and additions. In the 2D case, with an array of dimension N x N, the number of multiplications is in the order of $2N^3$ using a separable approach of computing 1D row and column DCTs. Specifically, for an 8 x 8 pixel array which is used in the JPEG family, 1024 multiplications and 896 additions are required. There have not been any significant improvements to reduce this computational overhead.

Even though the image data is an integer, their multiplication to cosine terms in the formula produces fractional numbers or real numbers because cosine values are fractional in nature until and unless they are integer multiples of Pi, which may not be the case. Since fractional numbers need infinite precision to store them exactly, they might produce errors in the reverse process, resulting in loss.

Another popular transformation is the wavelet transform. This is used, for example, in the JPEG2000 image compression standard. A mother wavelet is used to decompose the image data into frequency sub-bands, which in turn increases the redundancy in most of the sub-bands, thereby improving compression ratios. Used in their original form, the mother wavelets do not give integer-to-integer transformation but when used after a process called lifting, they come integer-to-integer transforms. This makes the entire process lossless but does not achieve a high compression ratio.

Color transformations also offer another compression technique. A commonly used color space is RGB. In RGB, every pixel is quantized by using a combination of Red, Green and Blue values. This format is popular among graphic designers, but is not ideal as a compression algorithm.

It is desirable to provide an image compression system which does not involve rigorous transforms, and complex calculations. It also has to be memory efficient and power efficient.

The true requirement is an image compression system which does not involve rigorous transforms, and complex calculations. It also has to be memory efficient and power efficient.

There are various image compression techniques presently available. A familiar few are JPEG, JPEG-LS, JPEG-2000, CALIC, FRACTAL and RLE.

JPEG compression is a trade-off between degree of compression, resultant image quality, and time required for compression/decompression. Blockiness results at high image compression ratios. It produces poor image quality when compressing text or images containing sharp edges or lines. Gibb's effect is the name given to this phenomenon - where disturbances/ripples may be seen at the margins of objects with sharp borders. It is not suitable for 2-bit black and white images. It is not resolution independent, and does not provide for scalability, where the image is displayed optimally depending on the resolution of the viewing device.

JPEG-LS does not provide support for scalability, error resilience or any such functionality. Blockiness still exist at higher compression ratios and it does not offer any particular support for error resilience, besides restart markers.

JPEG-2000 does not provide any truly substantial improvement in compression efficiency and is significantly more complex than JPEG, with the exception of JPEG-LS for lossless compression. The complexity involved in JPEG-2000 is higher for a lower enhancement in the compression ration and efficiency.

Although CALIC provides the best performance in lossless compression, it cannot be used for progressive image transmission as it implements a predictive-based algorithm that can work only in lossless/nearly-lossless mode. Complexity and computational cost are high.

All data compression techniques are based on the fundamental principle of Shannon's Information Theory. This theory states that there is a limit to the number of bits required to code a unique symbol, also known as entropy. This is given as the following equation:

$$\mathbf{H = - p_i \log_2 p_i}$$

Where Pi is the probability of occurrence of the symbol. The implication of this equation is that if a symbol occurs frequently, then this symbol contributes to repetition

and is designated a lower priority when compared to a symbol whose frequency of occurrence is less. This forms the basis for all the entropy coding or source coding schemes. A shorter codeword is given to more probable events. For example, the more frequently the symbol occurs, the shorter its codeword is.

Image data follows a Laplacian distribution. This means that the occurrence of each symbol is equiprobable. Thus, all the symbols require almost the same number of bits which results in very low compression ratios.

To achieve high compression, the image data stream is transformed from an even probability distribution in the original image to a probability distribution that has fewer symbols with a high frequency of occurrence and the remaining symbols with a relatively low frequency. This results in a significant reduction in bits per symbol and enhances the compression ratios.

Popular entropy encoders include run length encoders, Huffman, Shannon Fano, Limpel-Ziv and Arithmetic encoders. Most encoding techniques allot a minimum of at least one bit per symbol.

The results show that the choice of the "best" standard depends strongly on the application at hand.

Summary of the Invention

In a preferred aspect, there is provided a method for compressing image data of an image, comprising:

transforming the image data into a bit plane of first and second values;

comparing each image element with a previous image element and if they are both equal, recording a first value into the bit plane; and if they are not both equal, recording a second value into the bit plane; and

encoding repeating first and second values in the bit plane into a bit plane index;

wherein the compressed image is able to be decompressed lossless using the bit plane index and the bit plane.

The transformation may be a horizontal transformation, vertical transformation, predict transformation or a multidimensional transformation.

Each image element may be a pixel.

The first value may be a 1, and the second value may be a 0.

The bit-planes for the horizontal and vertical directions may be combined by binary addition to for a repetition coded compression bit-plane. Combining may be by binary addition, only the second values being stored for lossless reconstruction of the image. The result of the combining may be repetition coded compression data values. All other image data values may be able to be reconstructed using the repetition coded compression data values, and the bit planes for the horizontal and vertical directions.

Storage in bit planes may be in a matrix. A single mathematical operation may be performed for each element.

In another aspect, there is provided a system for compressing image data of an image, comprising:

> a transformation module to transform the image data into a bit plane of first and second values by comparing each image element with a previous image element and if they are both equal, recording a first value into the bit plane; and if they are not both equal, recording a second value into the bit plane; and an encoder to encode repeating first and second values in the bit plane into a bit plane index;

wherein the compressed image is able to be decompressed lossless using the bit plane index and the bit plane.

The bit-planes may contain information regarding the repetitions along horizontal and vertical directions. There may be further included the combining of the horizontal and vertical bit-planes by a binary addition operation to give a repetition coded compression bit-plane. There may also be included comparing the repetition coded compression bit-plane with the digital data matrix to obtain final repetition coded compression data values.

The method may further include storing and archiving the repetition coded compression data values along with the horizontal and vertical bit-planes.

The compression is preferably lossless. Alternatively, the method may further include compression by comparison with a threshold value to achieve lossy compression and a significantly higher compression ratio.

The method may be used for an application selected from: medical image archiving, medical image transmission, database system, information technology, entertainment, communications applications, and wireless application, satellite imaging, remote sensing, and military applications.

Brief Description of the Drawings

In order that the invention may be fully understood and readily put into practical effect, there shall now be described by way of non-limitative example only a preferred embodiment of the present invention, the description being with reference to the accompanying illustrative drawings in which:

Figure 1 illustrates the entire image compression system based on repetition coded compression on a hardware implementation;

Figure 2 is a sample grayscale image of a human brain, which is captured by magnetic resonance imaging ("MRI") to demonstrate the compression able to be achieved by repetition coded compression system;

Figure 3 is an enlarged image of a small region from Figure 2;

Figure 4 shows that the image of Figure 2 is made up of many pixels in grayscale;

Figure 5 shows a 36-pixel region within the sample MRI image of Figure 2;

Figure 6 shows the ASCII value equivalent of the image data values for the image of Figure 2;

Figure 7 shows the application of repetition coded compression along the horizontal direction in the image matrix;

Figure 8 shows the application of repetition coded compression along the vertical direction in the image matrix;

Figure 9 shows the combination of horizontal and vertical bit-planes by a binary addition operation;

Figure 10 shows the total memory required for the 36-pixel region before and after applying repetition coded compression;

Figure 11 shows the application of repetition coded compression to the entire image; and

Figure 12 shows the operational flow for the implementation of repetition coded compression.

Detailed Description of Preferred Embodiments

Image data is a highly correlated. This means that the adjacent data values in an image are repetitive in nature. Therefore, it is possible to achieve some compression out of this repetitive property of the image and then apply Huffman coding or other source coding schemes. Such a method would be very efficient.

High compression ratios can be achieved by combining existing data transforms and source encoders.

The human eye is more sensitive to luminance than color. Thus, chrominance luminance and value format offers another compression technique. This technique uses color transformations in image compression to generate visually lossless algorithms. Using lossy color transformation provides an effect equivalent to that of quantization of other techniques in the sense that it cannot resolve the difference between small values. That is, the same integer value is used for two different integer values with a small difference. As a result of this, repetition occurs at a 24-bit level. Increasing repetition in image data provides a high compression ratio. However, one drawback to this technique is that it is not reversible perfectly, that is, it is lossy. In other words, the decompressed image data is different from the original image data. The degree of difference is dependent upon the quality of compression and also the compression ratio. The adjustment of the quality may be user-defined by setting a quality parameter such that a very highly compressed visually lossless image is produced. By visually lossless we mean that the image data is technically lossy but to the human eye the image appears lossless.

A method for indexing a bit plane is provided which is flexible as it can be applied to a wide range of image types and formats. These image types include bi-level, grayscale,

8/16/24 bit color and medical images. The method is scalable as no change to the structure of the process is required for the various image types.

Bit plane indexing creates a redundant array of only zeros and ones. This improves the compression ratio without any loss or increase in the data set. This step is critical to obtain a high compression ratio to respond to speed.

In the bit plane indexing process, the raw original image data is decomposed to various types of bit planes. For example, these include horizontal, vertical or a combination of both, in an integer-to-integer matrix. A bit plane of zeros and ones is obtained along with the index of the image. The original image can be reconstructed perfectly losslessly with the index and the bit plane. The choice of which bit plane to use is dependent on the application or final product.

Bit plane indexing creates two arrays of codes. One array represents the index of the rearranged and sorted image. The second array is a set of zeroes and ones that form the bit plane.

Thus, the original image data is decomposed to one or more bit planes and stored along with an index of the image. The reconstruction is performed losslessly using the index and the bit plane.

In repetition coded compression ("RCC"), each element is compared with the previous element. If both of them are equal then a value of "1" is stored in a bit-plane. Otherwise a value of '0' is stored in the bit-plane. Only the difference value is stored in a matrix, instead of storing all the repeating values.

In a one-dimensional performance of the method, only one bit-plane is used to code the repetition. RCC horizontal transformation, RCC vertical transformation and RCC predict transformation are classified as RCC in one dimension.

<u>RCC Horizontal</u>

In RCC horizontal transformation only one bit-plane is used to code the repetition of values. That is, the bit-plane is in the horizontal direction only. In the RCC horizontal transformation, adjacent data elements, for example, pixels in the case of images, are

10

scanned in raster order (from left to right and then from top to bottom). If both adjacent data elements are equal, then a value of “1” is stored in the matrix or bit plane. Otherwise if they are not equal, a value of “0” is stored in the bit plane matrix. Only this different value is stored in the bit plane matrix instead of storing all the repeating values. Transforming the input data into a bit plane provides a greater amount of repetition than the original image data.

The RCC horizontal transformation only requires a logical mathematical comparison and no other mathematical calculation. The transformation falls within the integer-to-integer domain so as to maintain the lossless nature of the process. This process is ideal for images because a pixel is represented by 8 bits. When a logical transformation performed maps the pixel to another number, only 8 bits are required to be represented. This process preserves the lossless nature of the transform.

<u>RCC Vertical</u>

RCC vertical transformation is similar to the RCC horizontal transformation described except that image data is compared in a non-raster order. This transformation still preserves the lossless nature of the transform.

<u>RCC Predict</u>

RCC predict transformation compares two adjacent values in raster order. If the adjacent values are the same, then the value is stored in a bit plane matrix and gives a mapping value to the repeatedly occurring values and stores them in another data plane matrix. This method is suitable for medical images where different values repeat themselves, and these repetitions are replaced by a single mapping value and the actual value is stored in the data plane matrix. This transformation only performs logical transformations to the data and still preserves the lossless nature of the transform.

In a two-dimensional performance of the method, two bit-planes are used to code the repetitions in both the horizontal and the vertical directions. This is more efficient and gives a better compression ratio. RCC multidimensional transformation is classified as RCC in two dimensions.

RCC Multidimensional

A multidimensional bit plane performs a combination of the horizontal and vertical bit planes. In some cases, it is able to achieve improved compression ratios than just using either a horizontal or vertical bit plane. Firstly, the RCC horizontal transformation is performed and stores the generated bit plane as a horizontal bit plane. Next, a RCC vertical transformation is performed and the generated bit plane is stored as a vertical bit plane. A logical "OR" is performed on the two bit planes and stored as a lossless compressed multidimensional bit plane. A "NOT" operation is performed between the multidimensional bit plane and the original image matrix. Both the "OR" and "NOT" operations maintain the integrity of the image data and still preserves the lossless nature of the transform.

The compression system is based on a mathematical comparison of adjacent image data values. The comparison is performed between adjacent image data values in both the horizontal as well as vertical directions. The bit-planes formed as a result of the comparison in the horizontal and vertical directions are respectively combined by a binary addition method. After this the resultant bit-plane positions are called as RCC bit-planes. The zero values in the RCC bit-plane are stored for lossless reconstruction of the original image. For lossless reconstruction, they are the only values stored. The stored values correspond to the same locations in the original image matrix as zeros in the RCC bit-plane and are hereinafter called RCC data values. All the other image data values can be reconstructed by using the RCC data values, and the horizontal and vertical bit-planes.

Figure 1 illustrates the entire image compression system based on repetition coded compression on a hardware implementation. The analog image signals 12 are captured by the camera 10 and are converted into respective digital data 16 by a analog to digital converter 14. This digital data 16 is rearranged into a matrix of image data values by a reshaping block 18. The reshaped image matrix is stored in the embedded chip 20, which performs the entire repetition coded compression system. This therefore gives the compressed repetition coded compression data values 22 and also the bit-planes of data 24 for storage, archival and future retrieval 26.

Figure 2 is a sample image of the human brain which is captured by magnetic resonance imaging (MRI). This sample image may be used to demonstrate the compression achieved by repetition coded compression. It is a grayscale image.

Figure 3 zooms a small region from the sample MRI image of the human brain. This zoomed region may also be used for demonstrating the repetition coded compression system.

Figure 4 shows that the image is made up of lot of pixels in grayscale.

Figure 5 shows a 36-pixel region within the sample MRI image of the human brain.

Figure 6 shows the ASCII value equivalents of the image data values which are originally used for data storage. Each value requires eight bits (1 byte) of data memory. Currently, the 36-pixel region requires about 288 bits or 36 bytes of data memory. That data could be compressed and stored with only 112 bits after repetition coded compression.

Figure 7 shows the application of repetition coded compression along the horizontal direction in the image matrix. This results in the horizontal bit-plane and also the horizontal values stored.

Figure 8 shows the application of repetition coded compression along the vertical direction in the image matrix. This result in the vertical bit-plane, and also the vertical values stored.

Figure 9 shows the combination of horizontal and vertical bit-planes by a binary addition operation. This results in only five zero values which correspond to the final values stored from the original image matrix.

Figure 10 shows the total memory required for the 36-pixel region before and after applying repetition coded compression. The original memory requirement was 288 bits. After applying repetition coded compression the memory required was 112 bits. This is a great amount of compression.

Figure 11 shows the application of repetition coded compression to the entire image. The size is compressed to 44,000 bits from the original 188,000 bits.

Figure 12 shows an implementation of repetition coded compression. The image matrix 1201 is transposed 1202, encoded along the horizontal 1203 and vertical 1204 directions and the respective bit-planes 1205, 1206 are derived. Further compression is achieved by combining the horizontal and vertical bit-planes 1203, 1204 by a binary addition operation. This results in the repetition coded compression bit-plane 1207, which is logically inverted 1208 and compared 1209 with the original image matrix 1201 to obtain the final repetition coded compression data values 1210. The repetition coded compression data values 1210, together with the horizontal and vertical 1206bit-planes are stored in a data memory 1211 for archival and future retrieval.

The coded data can be further compressed by Huffman coding. This compression of the image data is achieved using the repetition coded compression system. This system is fast as it does not make use of complex transform techniques. The method may be used for any type of image file. In the example given above, the system is applied only for grayscale images. It may be applied to color images.

The system of repetition coded compression of images may be applied to fields such as, for example, medical image archiving and transmission, database systems, information technology, entertainment, communications and wireless applications, satellite imaging, remote sensing, military applications.

The preferred embodiment of the present invention is based on a single mathematical operation and requires no multiplication for its implementation. This results in memory efficiency, power efficiency, and speed, in performing the compression. Because of the single mathematical operation involved, the system is reversible and lossless. This may be important for applications which demand zero loss. The compression ratios may be significantly higher than existing lossless compression schemes.

If the application permits a lossy compression system, a modification is made to the mathematical operation so that a certain amount of loss is observed in the compression, thereby resulting in higher compression ratios. This lossy compression system would find great applications in entertainment and telecommunication systems.

Whilst there has been described in the foregoing description a preferred embodiment of the present invention, it will be understood by those skilled in the technology that many variations or modifications in details of design, constructions or operation may be made without departing from the present invention.

A method for compressing image data of an image, comprising:

transforming the image data into a bit plane of first and second values;

comparing each image element with a previous image element and if they are both equal, recording a first value into the bit plane; and if they are not both equal, recording a second value into the bit plane; and

encoding repeating first and second values in the bit plane into a bit plane index;

the compressed image is able to be decompressed lossless using the bit plane index and the bit plane.

The transformation is a repetition coded compression horizontal transformation, repetition coded compression vertical transformation, repetition coded compression predict transformation or a repetition coded compression multidimensional transformation.

Each image element is a pixel.

The first value is a 1, and the second value is a 0.

The repetition coded compression horizontal transformation, repetition coded compression vertical transformation, repetition coded compression predict transformation, a single bit plane is used to store the values.

The repetition coded compression multidimensional transformation, comparison is in both horizontal and vertical directions, a separate bit plane being used for each direction.

The bit-planes for the horizontal and vertical directions are combined by binary addition to form a repetition coded compression bit-plane.

15

The combining is by binary addition, only the second values being stored for lossless reconstruction of the image.

The result of the combining is repetition coded compression data values, all other image data values being able to be reconstructed using the repetition coded compression data values, and the bit planes for the horizontal and vertical directions.

Storage in bit planes is in a matrix.

A single mathematical operation is performed for each image element.

A system for compressing image data of an image, comprising:
a transformation module to transform the image data into a bit plane of first and second values by comparing each image element with a previous image element and if they are both equal, recording a first value into the bit plane; and if they are not both equal, recording a second value into the bit plane; and
an encoder to encode repeating first and second values in the bit plane into a bit plane index;
wherein the compressed image is able to be decompressed lossless using the bit plane index and the bit plane.

The method is used for an application selected from the group consisting of: medical image archiving, medical image transmission, database system, information technology, entertainment, communications applications, and wireless application, satellite imaging, remote sensing applications.

Dated This 13th Day of April 2004.

(S.D.AHUJA)
Of D.P.AHUJA&CO
APPLICANT'S AGENT

MATRIXVIEW TECHNOLOGIES (INDIA) PRIVATE LIMITED 12 SHEETS

NO. 357 /CHE/2004 SHEET 1

PROVISIONAL

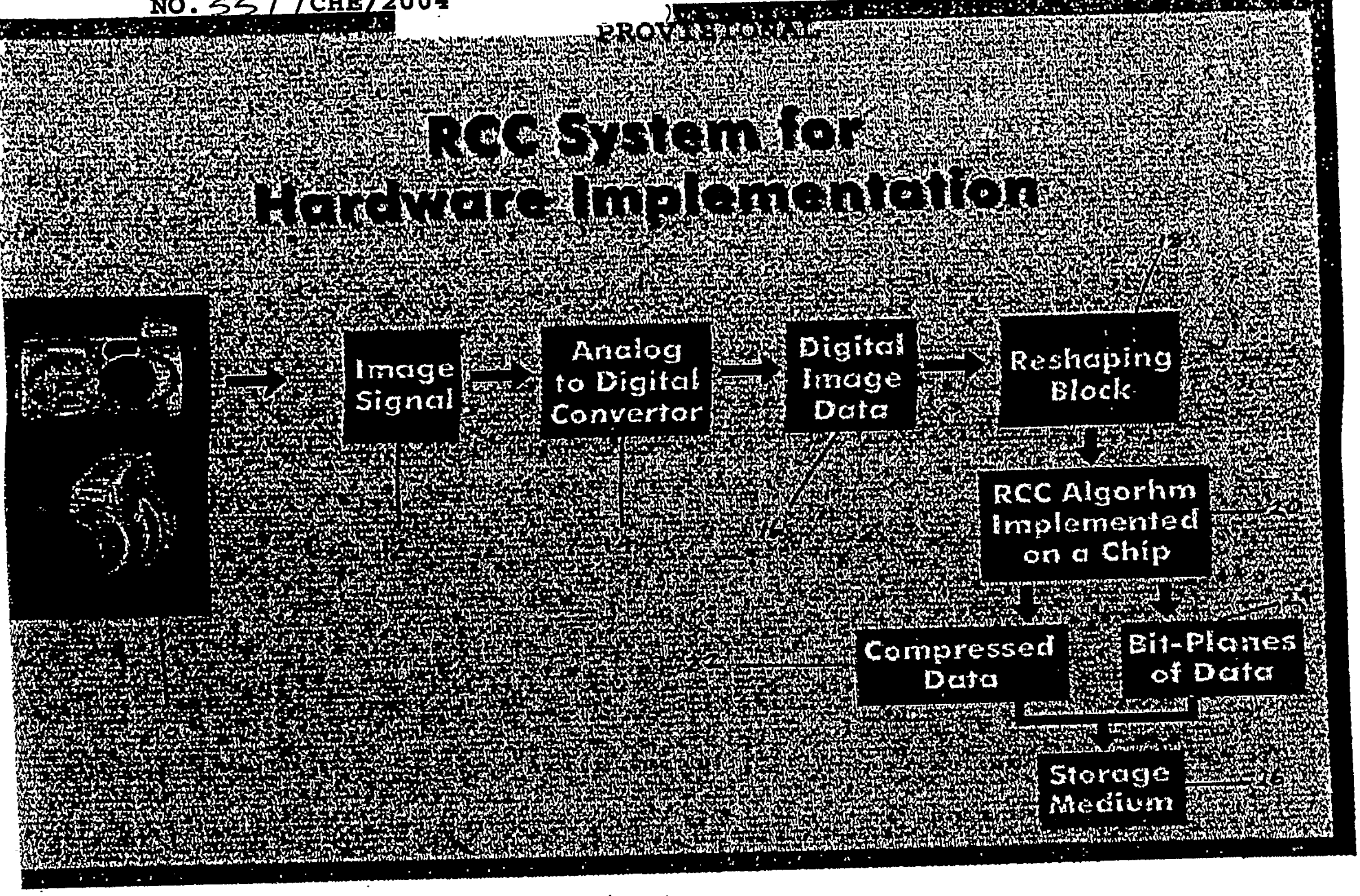

Figure 1

(SOUMEN MUKHERJEE)
of D. P. AHUJA & CO.
APPLICANTS' AGENT

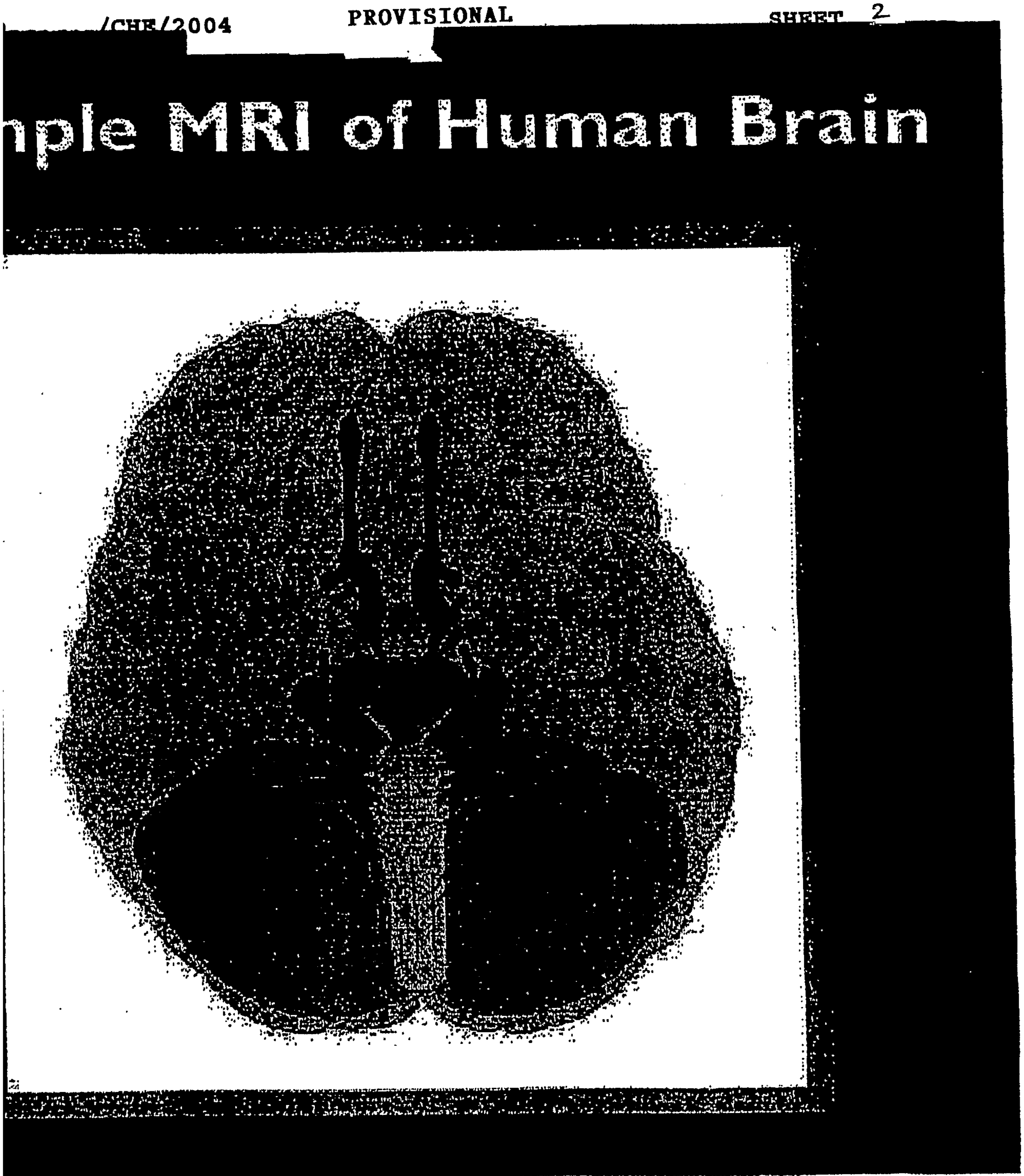

Figure 2

(SOUMEN MUKHERJEE)
of D. P. AHUJA & CO.
APPLICANTS' AGENT

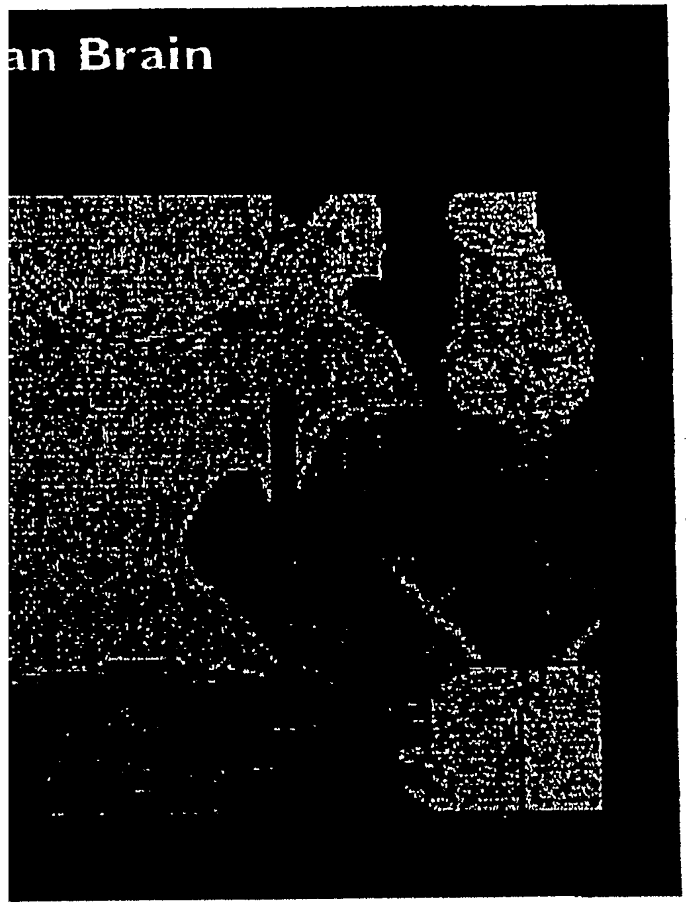

Figure 3

(SOUMEN MUKHERJEE)
of D. P. AHUJA & CO.
APPLICANTS' AGENT

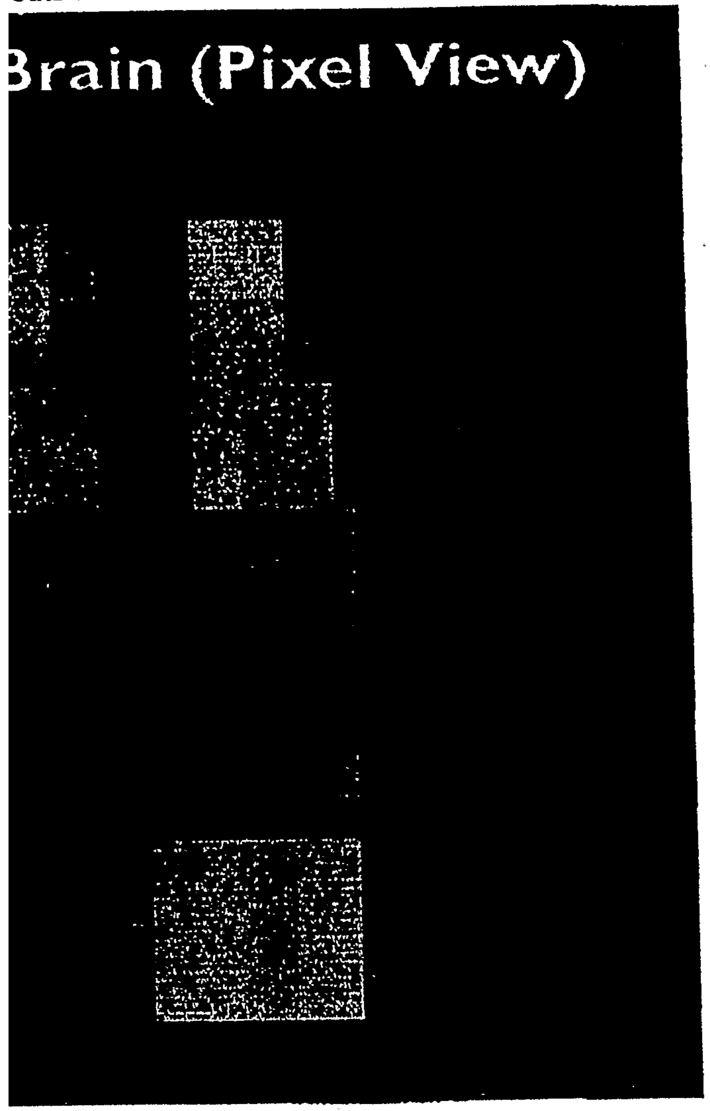

Figure 4

(SOUMEN MUKHERJEE)
of D. P. AHUJA & CO.
APPLICANTS' AGENT

OGIES (INDIA) PRIVATE LIMITED
PROVISIONAL

12 SHEETS
SHEET 5

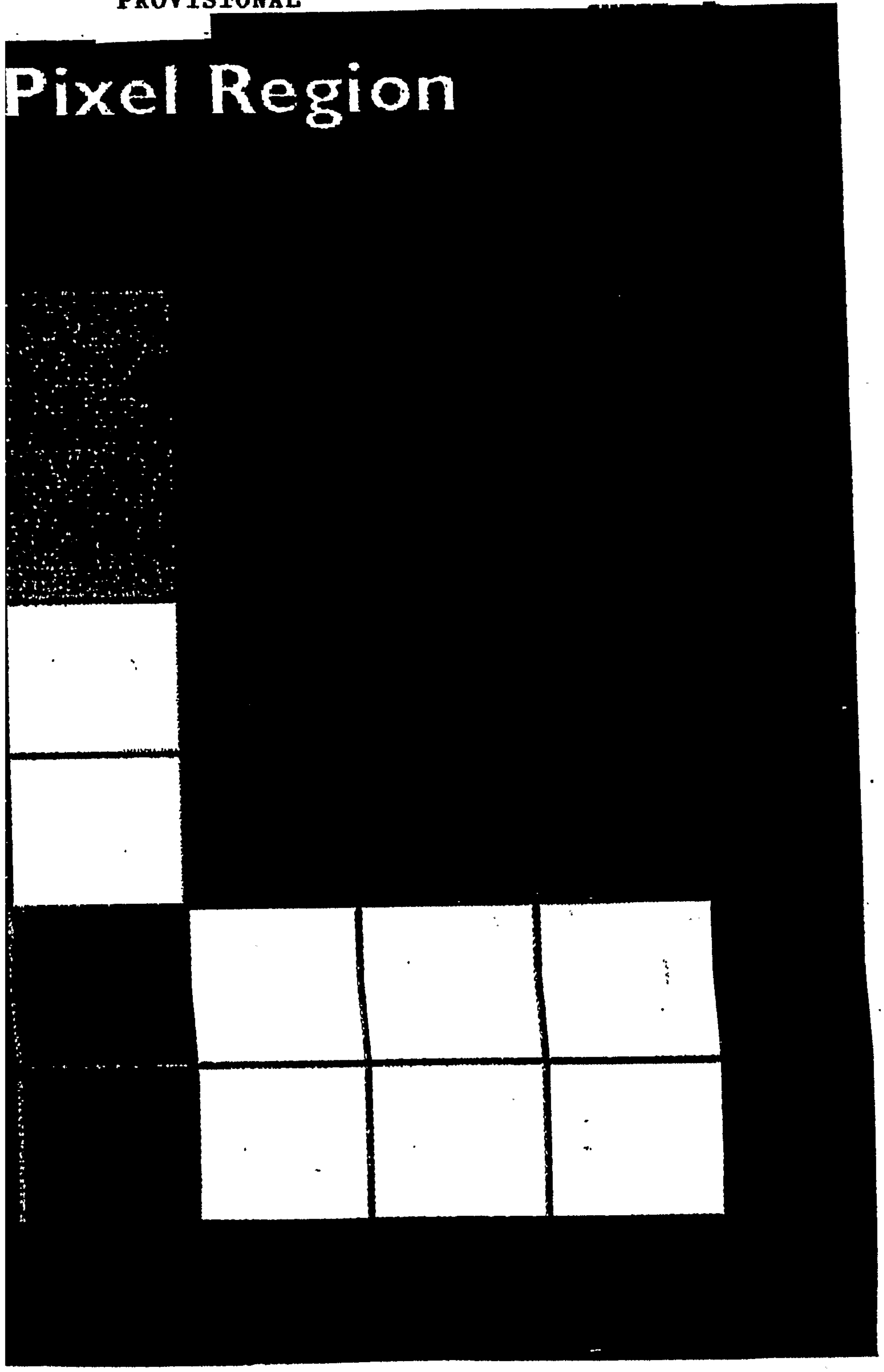

FIGURE 5

(SOUMEN MUKHERJEE)
of D. P. AHUJA & CO.
APPLICANTS' AGENT

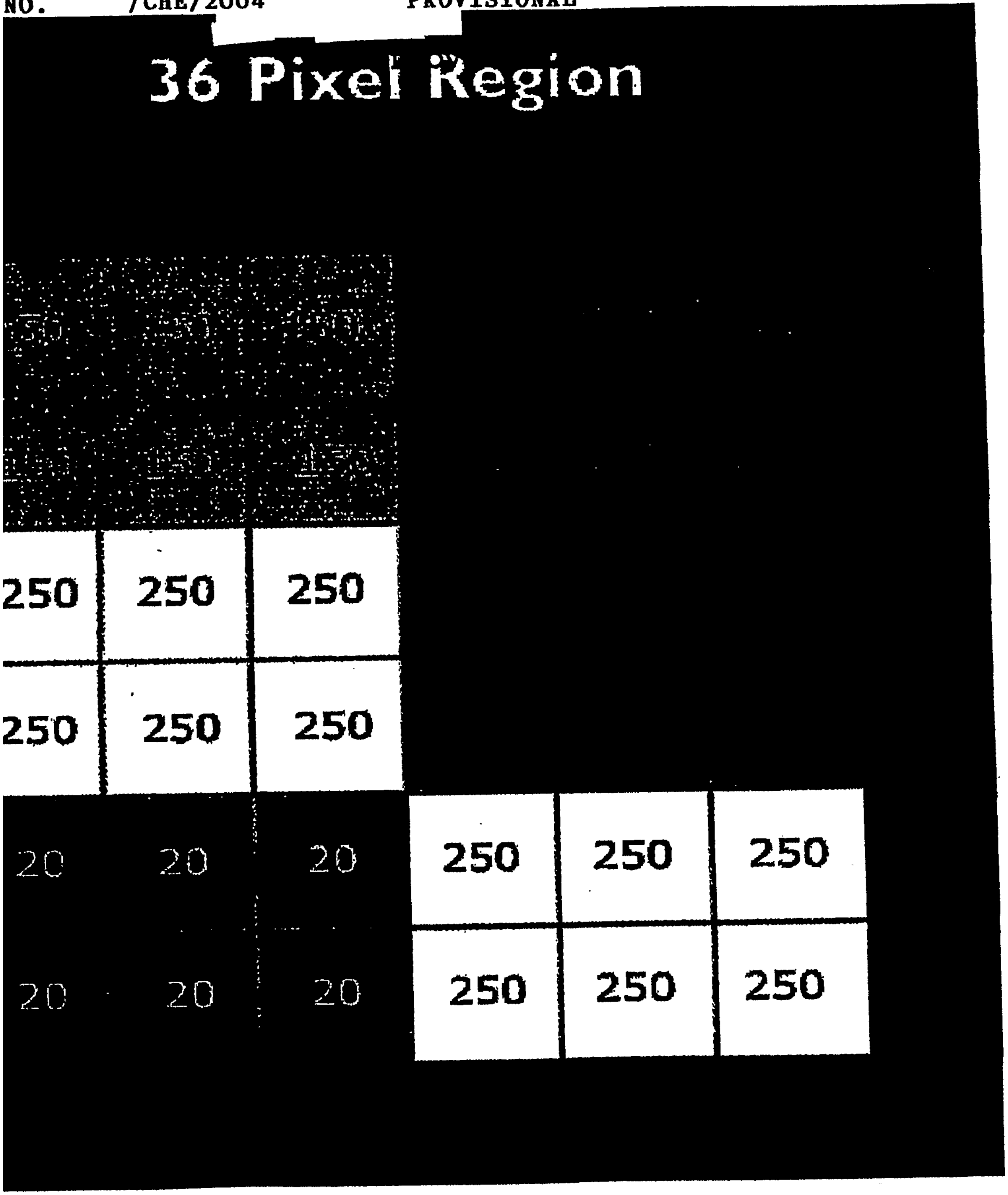

FIGURE 6

(SOUMEN MUKHERJEE)
of D. P. AHUJA & CO.
APPLICANTS' AGENT

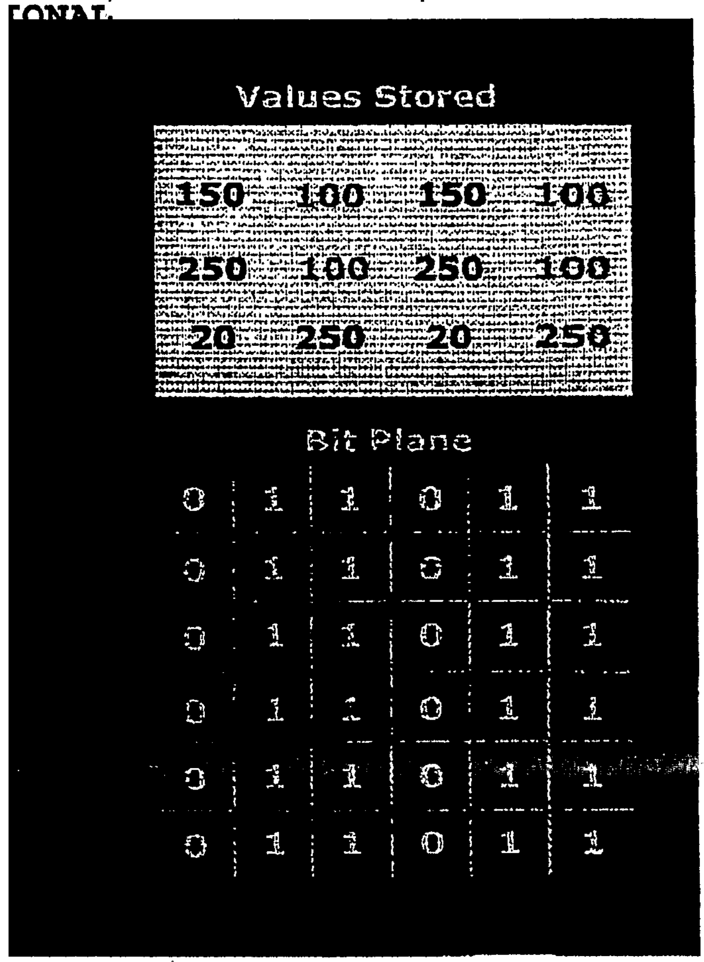

Figure 7

(SOUMEN MUKHERJEE)
of D. P. AHUJA & CO.
APPLICANTS' AGENT

PROVISIONAL

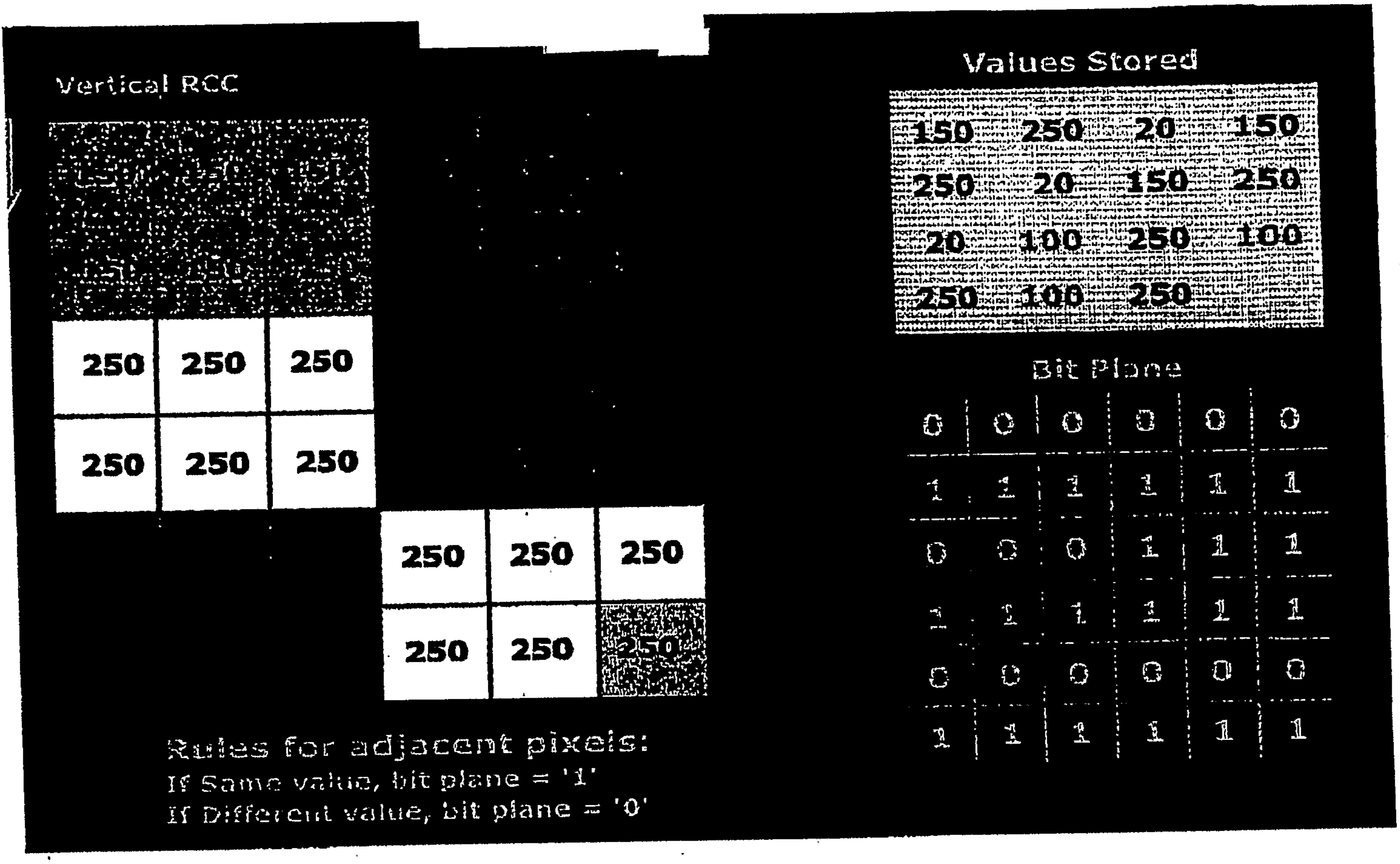

Figure 8

(SOUMEN MUKHERJEE)
of D. P. AHUJA & CO.
APPLICANTS' AGENT

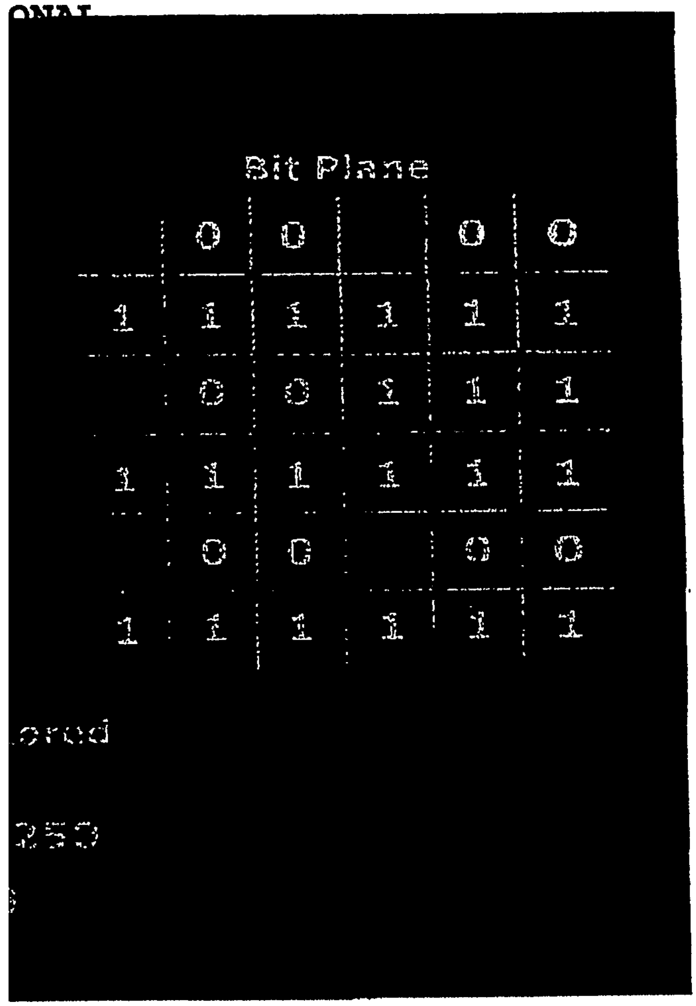

Figure 9

(SOUMEN MUKHERJEE)
of D. P. AHUJA & CO.
APPLICANTS' AGENT

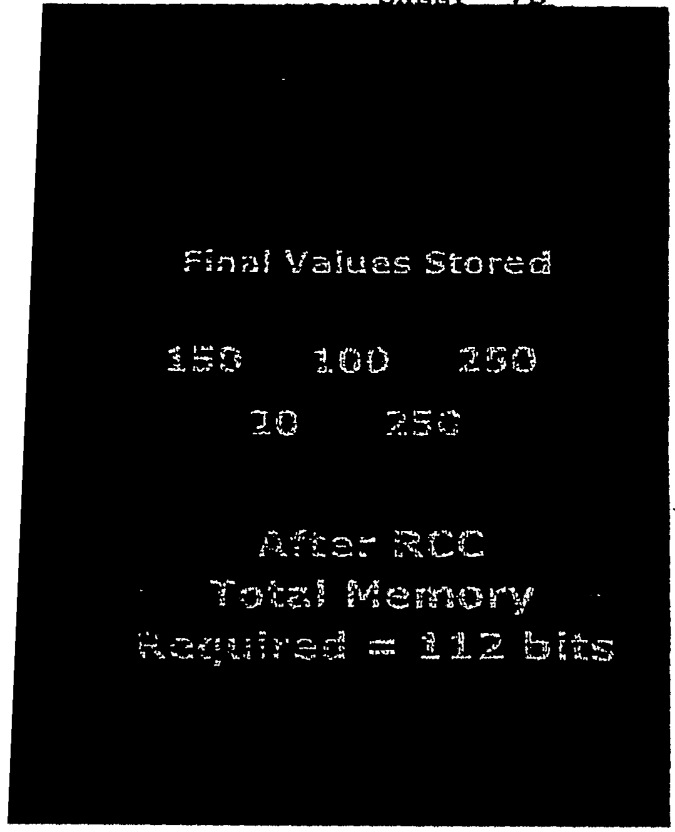

Figure 10

(SOUMEN MUKHERJEE)
of D. P. AHUJA & CO.
APPLICANTS' AGENT

PROVISIONAL

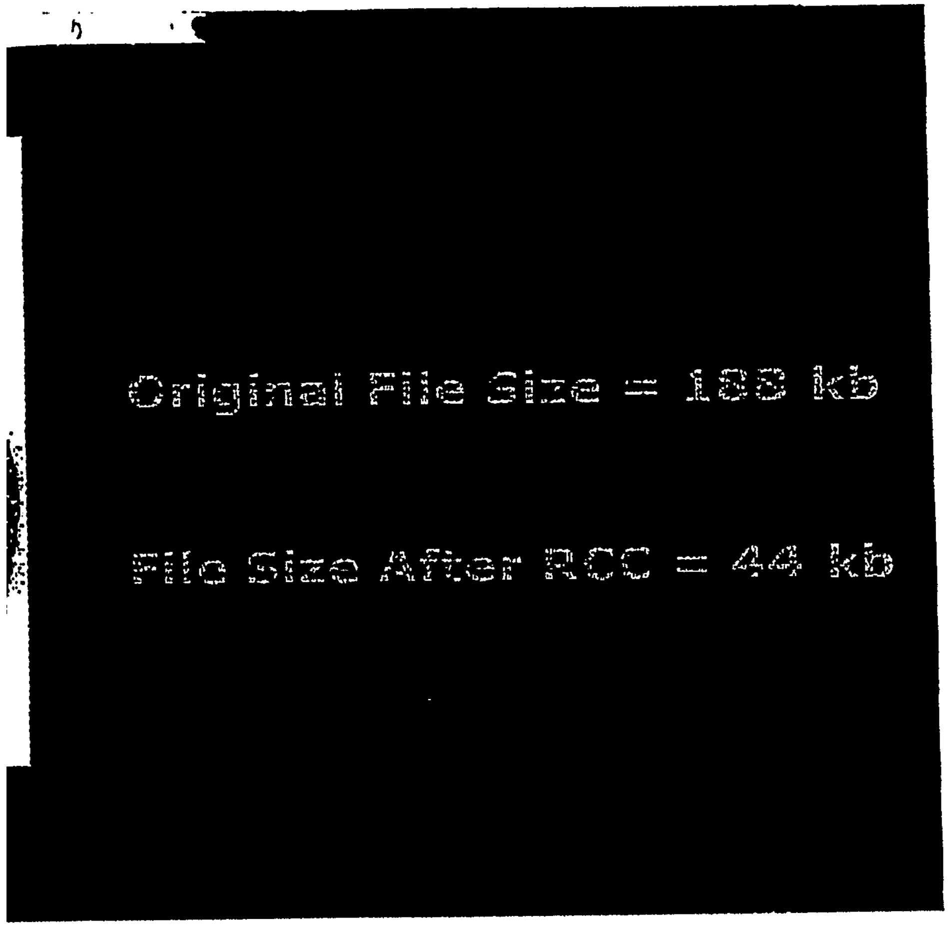

Figure 11

(SOUMEN MUKHERJEE)
of D. P. AHUJA & CO.
APPLICANTS' AGENT

MATRIXVIEW TECHNOLOGIES (INDIA) PRIVATE LIMITED 12 SHEETS

NO. /CHE/2004 SHEET 12

PROVISIONAL

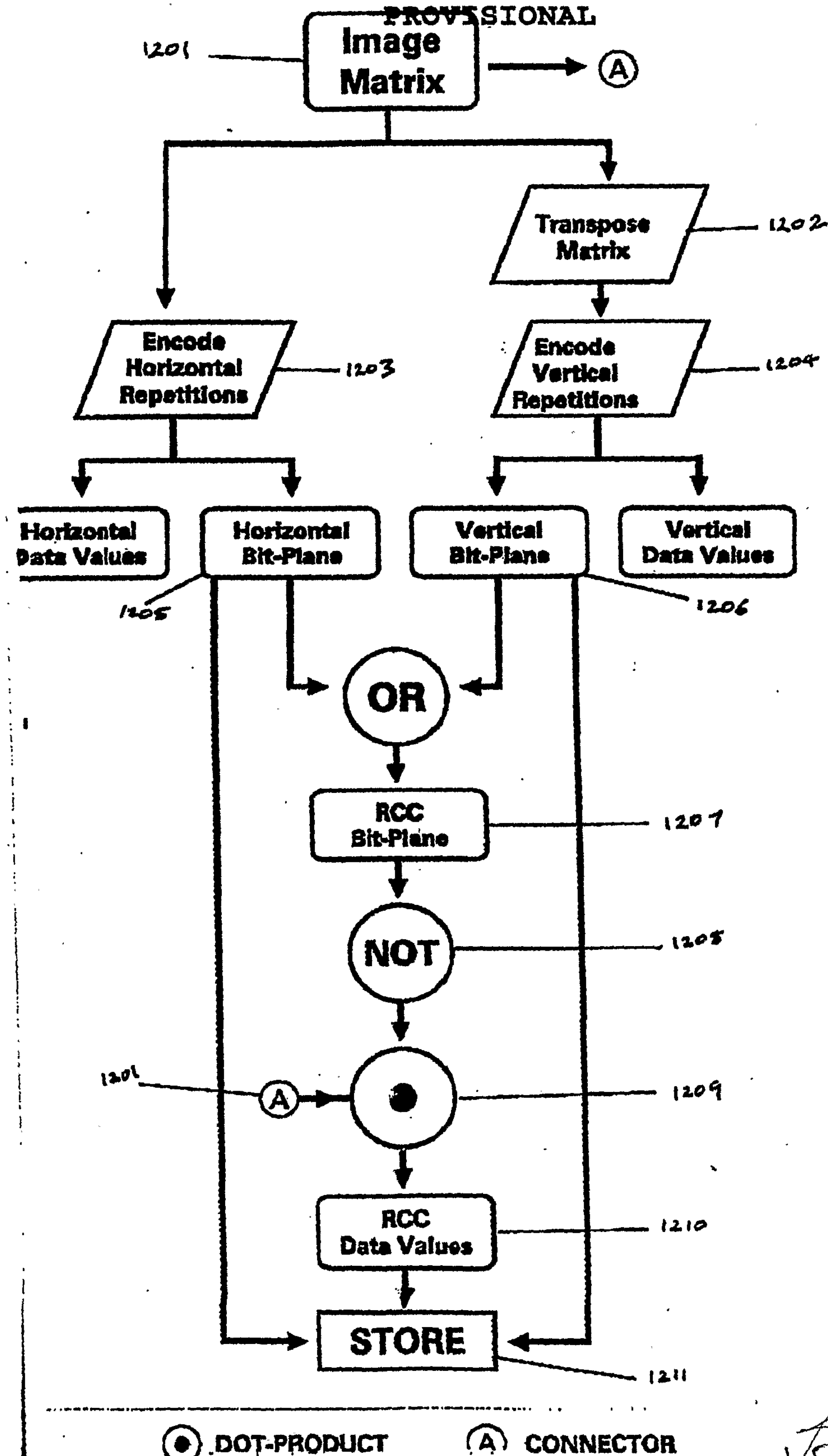

FIGURE 12

(SOUMEN MUKHERJEE)
of D. P. AHUJA & CO.
APPLICANTS' AGENT